

---

**<SJTU>**

---

**< Honeycomb>**

软件需求规约

版本 **<1.0>**

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

## 修订历史记录

日期	版本	说明	作者
<日/月/年>	<x.x>	<详细信息>	<姓名>
30/5/208	0.1	初步起草文件	谢添翼、曹金坤
1/6/2018	0.2	修改文件细节，添加用例说明	马轲、谢添翼、朱锦昊
2/6/2018	1.0	修改文件错误，小组最后核对	马轲、谢添翼、朱锦昊、曹金坤

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

## 目录

1. 简介	5
1.1 目的	5
1.2 定义、首字母缩写词和缩略语	5
1.3 参考资料	5
2. 整体说明	5
3. 具体要求	6
3.1 功能	6
3.1.1 <Use case 图>	6
3.1.2 <登录 规约>	6
3.1.3 <注册 规约>	7
3.1.4 <搜索框搜索论文 规约>	7
3.1.5 <网络图搜索论文 规约>	8
3.1.6 <浏览论文 规约>	8
3.1.7 <评论 规约>	8
3.1.8 <收藏 规约>	9
3.1.9 <管理个人账号 规约>	9
3.1.10 <管理个人信息 规约>	9
3.1.11 <管理个人收藏 规约>	10
3.1.12 <管理个人网络图 规约>	10
3.1.13 <管理个人历史记录 规约>	10
3.2 易用性	11
3.2.1 <用户操作难度>	11
3.2.2 <界面>	11
3.2.3 <导航>	11
3.2.4 <图形标准>	11
3.3 可靠性	11
3.3.1 <可用时间>	11
3.3.2 <平均故障间隔时间>	11
3.3.3 <平均修复时间>	12
3.3.4 <精确率>	12
3.3.5 <最高缺陷率>	12
3.3.6 <错误>	12
3.4 性能	12
3.4.1 <对事务的响应时间>	12
3.4.2 <吞吐量>	12
3.4.3 <容量>	12
3.4.4 <降级模式>	12
3.4.5 <资源利用情况>	12
3.5 可支持性	12
3.5.1 <编码标准>	12
3.5.2 <命名约定>	12
3.5.3 <访问权限>	13
3.6 设计约束	13
3.6.1 <设计约束一>	13

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

3.6.2 <设计约束二>	13
3.7 接口	13
3.7.1 用户界面	13
3.7.2 硬件接口	14
3.7.3 软件接口	14
3.7.4 通信接口	14

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

# 软件需求规约 (简化版)

## 1. 简介

### 1.1 目的

本文档旨在定义 paper reference net 应用的总体要求，建立相应逻辑模型，明确用户需求，并且提供应用的功能说明和实现的基本技术路径，此外还说明了工程的部分非功能性需求，并且对设计规范和其他约束进行解释。

### 1.2 定义、首字母缩写词和缩略语

SHC: "Scholar Honeycomb" ("学术蜂巢") 首字母的缩写词

### 1.3 参考资料

《软件工程原理》

## 2. 整体说明

产品总体效果：提供一个可供用户完成登录查询信息，基于论文关系网络进行探索进而寻找目标论文的平台，该平台的基础版本是基于 web 进行实现的。

产品功能：

1. 用户登录/注册：保存用户个人信息，提供用户进一步详细查询信息及构建个人论文网络的功能
2. 论文网络图展示：提供基于论文联系网络，方便用户探索查找
3. 论文搜索：提供用户论文搜索的功能，该功能会继承多种学术搜索引擎
4. 论文评论、收藏：提供用户评论论文或者收藏论文的功能
5. 个人历史记录：提供用户查看论文浏览历史记录的功能
6. 个人网络图：提供用户构建私人论文网络图的功能，用户可以根据自己的需要对自建文件或者网络文档的知识图谱进行构建

用户特征：Paper Reference Net 面向广大需要查询阅读论文的科研工作人员以及其他需要对不同领域专业文档和学术论文进行检索的潜在用户

约束：

1. 能够利用爬虫或许网上的论文信息和对应的文档链接
2. 能够建立推荐算法模型，基于用户现有的使用记录，进行相关内容的推荐
3. 对于涉及到的用户关系和论文作者、机构的 reference/citation 和隶属关系，需要基于社交网络的算法模型进行实践

假设与依赖关系：根据该工程的开发实际和应用场景，需要对工程做出的假设和依赖有：

1. 选择 Java+Vue.js 作为我们的开发起点，这基于以下依赖和假设：
  - a) 小组成员均对 Vue.js 有一定的使用经验，利于开发中的协同作业。此外，Vue.js 可以完成计划中的数据可视化、复杂交互逻辑等前端作业；
  - b) 基于 Java 可以获取足够的现有工具，完成对于前后端的粘合以及对数据库的操作；

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

## 2. 部署环境假设和依赖:

对于该基于 web 的工程，我们选用的云虚拟主机可以支持网站运行要求的全部资源要求；

## 3. 数据来源依赖:

网站的论文索引系统的查询接口直接对接现有的学术搜索引擎的数据检索接口，网站缓存的学术论文和文档资料可以通过上述机构的现有 api 或者自主编写的爬虫系统获得。

需求子集:

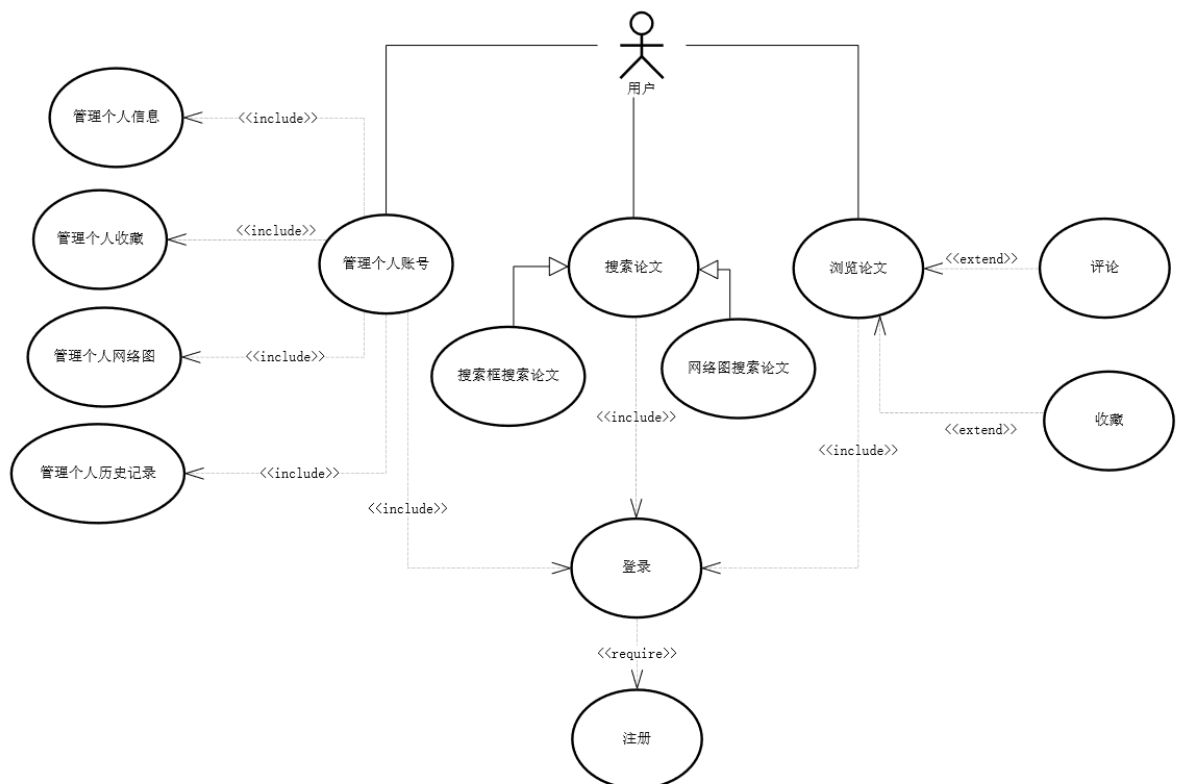
网站安全、稳定，对于用户的操作在合理范围内给出正确的回应；网站的数据来源可靠且完整；网站具有一定的社交功能，对于用户的评价和自己设置的知识图谱等内容不设置存储空间的限制；网站提供便捷的错误反馈功能。

# 3. 具体需求

## 3.1 功能

对于“HoneyComb”工程的功能说明，自然语言的描述已经在之前的章节中提供，该节中会提供基于 PowerDesigner 的用例建图进行功能具体操作流和逻辑的说明，并提供详细的用例规约说明。

### 3.1.1 <Use case 图>



### 3.1.2 <登录 规约>

用例编号:	SHC001	用例名称:	登录
描述:	用户在使用其他功能前应在 web 浏览器登录		

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

执行者:	用户
前置条件:	
基本流:	<ol style="list-style-type: none"> <li>1. 用户单击“登录”按钮</li> <li>2. 系统显示登录界面</li> <li>3. 用户输入登录信息</li> <li>4. 用户点击“登录”按钮</li> <li>5. 系统将用户填写的表单发送给服务端，服务端进行验证，返回验证成功信息</li> <li>6. 显示登录成功，跳转至首页</li> </ol>
备选流:	<ol style="list-style-type: none"> <li>4a. 用户单击“注册”按钮 系统进入“注册”用例</li> <li>4b. 用户点击“返回首页”按钮 系统跳转至首页，用例结束</li> <li>5a. 系统提示“用户名或密码错误” 返回第 2 步</li> </ol>
非功能需求:	系统响应用户时间不超过 2 秒

### 3.1.3 <注册 规约>

用例编号:	SHC002	用例名称:	注册
描述:	用户注册账号		
执行者:	用户		
基本流:	<ol style="list-style-type: none"> <li>1. 用户单击“注册”按钮</li> <li>2. 系统显示注册界面</li> <li>3. 用户填写注册信息</li> <li>4. 用户单击“提交”注册按钮</li> <li>5. 系统将表单信息发送给服务端，服务端进行验证，返回验证成功信息</li> <li>6. 显示注册成功，跳转至登录界面</li> </ol>		
备选流:	<ol style="list-style-type: none"> <li>1-4a. 用户点击“返回首页”按钮 系统跳转至首页，用例结束</li> <li>4b. 用户单击“取消”按钮 系统显示首页</li> <li>5a. 两次输入密码不一致 系统显示“密码与确认密码不一致”，返回 2</li> <li>5b. 系统提示“用户名已被注册” 返回 3</li> <li>5c. 手机号不符合规范 系统显示“不是有效手机号”，返回 3</li> <li>5d. 邮箱不符合规范 系统显示“不是有效邮箱”，返回 3</li> </ol>		
非功能需求:	系统响应用户时间不超过 2 秒		

### 3.1.4 <搜索框搜索论文 规约>

用例编号:	SHC003	用例名称:	搜索框搜索论文
描述:	用户通过使用搜索框进行关键字搜索寻找论文		
执行者:	用户		
后置条件:	用户的查看详情操作和查询历史被系统记录		

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

基本流:	1. 用户点击搜索图标按钮 2. 客户端显示搜索框 3. 用户输入关键字 4. 系统将表单发送至服务端，服务端进行搜索，并返回搜索结果 5. 系统显示搜索结果以及论文网络图
备选流:	1-3a. 返回 跳转至其他界面，用例结束
非功能需求:	系统响应用户时间不超过 2 秒

### 3.1.5 <网络图搜索论文 规约>

用例编号:	SHC004	用例名称:	网络图搜索论文
描述:	用户通过论文网络关系图来寻找目标论文		
执行者:	用户		
后置条件:	用户的查看详情操作和查询历史被系统记录		
基本流:	1. 用户点击显示出的论文网络图中某个元素 2. 系统显示论文概况以及扩展后的网络图		
备选流:	1-2a. 返回 跳转至其他界面，用例结束		
非功能需求:	系统响应用户时间不超过 2 秒		

### 3.1.6 <浏览论文 规约>

用例编号:	SHC005	用例名称:	浏览论文
描述:	用户浏览论文简介		
执行者:	用户		
前置条件:	用户需登录系统		
基本流:	1. 界面显示论文列表 2. 用户浏览论文简介 3. 用户点击某一篇论文 4. 系统发送请求至服务器，服务器返回相应内容，系统跳转至论文详细界面 5. 用户查看论文详细内容		
备选流:	1-3a. 用户点击“返回首页”按钮 系统跳转至主页面，用例结束 5a. 用户点击“评论”按钮 系统进入评论用例 5b. 用户点击“收藏”按钮 系统进入收藏用例		
非功能需求:	系统响应用户时间不超过 2 秒		

### 3.1.7 <评论 规约>

用例编号:	SHC006	用例名称:	评论
描述:	用户评论论文		
执行者:	用户		



<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

前置条件:	用户需登录系统
基本流:	1. 显示评论界面 2. 用户在输入框中输入评论, 点击发表 3. 系统将相关评论信息发送至服务端, 服务端记录相关操作 4. 系统刷新评论界面
非功能需求:	系统响应用户时间不超过 2 秒

### 3.1.8 <收藏 规约>

用例编号:	SHC007	用例名称:	收藏
描述:	用户收藏论文		
执行者:	用户		
前置条件:	用户需登录系统		
基本流:	5. 用户点击“收藏”按钮 6. 系统将相关收藏信息发送至服务端, 服务端记录相关操作, 更新用户个人信息 7. 系统刷新界面		
非功能需求:	系统响应用户时间不超过 2 秒		

### 3.1.9 <管理个人账号 规约>

用例编号:	SHC008	用例名称:	管理个人账号
描述:	用户查看个人信息, 并可有选择的修改个人信息		
执行者:	用户		
前置条件:	用户需登录系统		
基本流:	1. 用户点击“个人空间”按钮 2. 系统将请求发送至服务端, 服务器接受请求并返回用户信息 3. 系统接收用户信息, 并生成显示界面 4. 用户查看个人空间		
备选流:	2a. 用户无权访问个人空间 (如: 未登录), 服务器返回对应错误信息用例结束 4a. 用户点击“个人信息”按钮, 进入管理个人信息用例 4b. 用户点击“个人收藏”按钮, 进入管理个人收藏用例 4c. 用户点击“个人网络图”按钮, 进入管理个人网络图用例 4d. 用户点击“个人历史记录”按钮, 进入管理个人历史记录用例		
非功能需求:	系统响应用户时间不超过 2 秒		

### 3.1.10 <管理个人信息 规约>

用例编号:	SHC009	用例名称:	管理个人信息
描述:	用户查看个人信息, 并可有选择的修改个人信息		
执行者:	用户		
前置条件:	用户需登录系统		
基本流:	1. 用户在查看个人信息页面点击“修改” 2. 系统根据已有用户信息生成修改界面 3. 用户修改个人信息 4. 用户点击“修改”按钮 5. 系统将修改信息发送至服务端, 服务端验证信息正确性, 若合法则更新用户信息库, 并返回修改结果 6. 系统根据修改结果重新生成个人信息界面		

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

备选流:	1-4a. 用户点击“返回首页”按钮 系统跳转至主页面, 用例结束 5a. 服务端检验到用户信息不合法, 返回错误信息 系统提示“修改信息不合法”, 回到第 1 步
非功能需求:	系统响应用户时间不超过 2 秒

### 3.1.11 <管理个人收藏 规约>

用例编号:	SHC010	用例名称:	管理个人收藏
描述:	用户查看个人收藏, 并可有选择的修改个人信息		
执行者:	用户		
前置条件:	用户需登录系统		
基本流:	1. 用户点击“个人收藏” 2. 系统发送用户信息至服务端, 服务端返回用户对应收藏内容 3. 系统根据服务端返回信息, 生成相应收藏界面 4. 用户点击某条收藏条目旁“取消收藏”按钮, 系统发送请求至服务端 5. 服务端取消收藏记录, 更新用户信息库, 返回执行结果 6. 系统收到反馈, 去掉对应收藏条目		
非功能需求:	系统响应用户时间不超过 2 秒		

### 3.1.12 <管理个人网络图 规约>

用例编号:	SHC011	用例名称:	管理个人网络图
描述:	用户查看个人网络图, 并可有选择的修改个人网络图		
执行者:	用户		
前置条件:	用户需登录系统		
基本流:	1. 用户点击“个人网络” 2. 系统发送用户信息至服务端, 服务端返回用户对应个人创建的网络列表 3. 用户点击某个网络 4. 系统显示该网络详情 5. 用户编辑该网络信息 6. 系统将修改信息发送至服务端, 服务端验证信息正确性, 若合法则更新用户个人网络, 并返回修改结果 7. 系统根据修改结果重新生成网络详情并显示		
备选流:	3-7a. 用户点击“返回” 返回第 2 步 3b. 用户点击“新建网络” 系统产生一张空的网络图并显示 6b. 服务端检验到用户信息不合法, 返回错误信息 系统提示“修改信息不合法”, 返回第 2 步		
非功能需求:	系统响应用户时间不超过 2 秒		

### 3.1.13 <管理个人历史记录 规约>

用例编号:	SHC012	用例名称:	管理个人历史记录
描述:	用户查看个人历史记录, 并可有选择的修改个人历史记录		
执行者:	用户		
前置条件:	用户需登录系统		

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

基本流:	<ol style="list-style-type: none"> <li>1. 用户点击“个人历史记录”按钮</li> <li>2. 系统发送用户信息至服务端，服务端返回用户对应历史记录内容</li> <li>3. 系统根据服务端返回信息，生成相应历史记录界面</li> <li>4. 用户点击某条历史记录旁“删除”按钮，系统发送请求至服务端</li> <li>5. 服务端取消历史记录，更新用户信息库，返回执行结果</li> <li>6. 系统收到反馈，去掉对应历史记录</li> </ol>
非功能需求:	系统响应用户时间不超过 2 秒

### 3.2 易用性

应用涉及的用户主要为包括教授、学生等在内的各类研究学者，应用应满足能为这些用户带来便捷、友好的使用体验

#### 3.2.1 <用户操作难度>

对于掌握计算机基本操作的用户，其初次使用 HoneyComb 网站的难度应该不高于：

- 一个用户应该可以在平均 2 分钟最多 3 分钟的时间掌握根据网络图的基本查询操作；
- 因为文件的查询逻辑和通用搜索引擎类似，不存在额外的学习负担；
- 用户应在平均 5 分钟最多 10 分钟的时间内掌握私人网络图的构建方式，对于熟悉 MindMap 和 UML 图例的用户，不存在额外的学习成本；
- 因为论文心得提交和评价等社交功能类似于当下的微博和朋友圈，所以具有通用和被大家广泛熟悉的操作逻辑，用户可以在几分钟内快速熟悉。

#### 3.2.2 <界面>

- 界面应风格统一，有良好的视觉体验；
- 网站界面具有简洁的设计逻辑，对于小构件的设计遵循类似于 Google MaterialDesign 的块状设计，对于大的界面设计较多的使用圆角和圆润的设计理念；
- 界面突出功能性组件，使使用者可以便利地获取操作逻辑的入口。

#### 3.2.3 <导航>

- 当用户第一次登陆使用“HoneyComb”网站时，对其提供一个基于滑动框的使用说明，滑动框应当位于页面的中央，占据约 50%的界面控件；
- 当用户第一次使用某功能接口时提供简洁的功能介绍和流程说明；
- 当用户第一次进入某界面，并且在该界面停留了较久时间没有进行有效操作时，再次提供简介的操作流程说明导航。

#### 3.2.4 <图形标准>

- 图形使用符合 Microsoft 的 GUI 标准要求；
- 图形设计中广泛应用 Google MaterialDesign 的图形设计规范，追求统一的图形界面视觉效果。

### 3.3 可靠性

#### 3.3.1 <可用时间>

在不发生地震、台风等不可抗事故前提下，在网站托管的虚拟服务器提供者设备不生长时间故障的前提下，一年中 Honeycomb 平均可用时间百分比应至少达到 99.00%

#### 3.3.2 <平均故障间隔时间>

应用的平均故障间隔时间应不小于 7 天。

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

### 3.3.3 <平均修复时间>

应用修复的平均修复时间应不大于 2 小时，对于因为后台服务器故障和不可抗因素导致的宕机等问题，可能会通过更换服务器等方法解决。此时的故障恢复时间不超过 48 小时。

### 3.3.4 <精确率>

因为网站的论文数据和第三方提供的代码文件等数据依赖于包括第三方 api 和爬虫等非自主的数据获取通道，同时，考虑到网站托管设备的性能和存储空间限制，对于网站中保存的数据信息无法保证数据的完全实时。但是，对于在线获取的数据文件，精确度应不小于 95.00%，而对于依赖于数据缓存和第三方数据获取通道的数据文件，数据的滞后时间不超过 24 小时。

### 3.3.5 <最高缺陷率>

- 平均每千行代码的错误数目应不超过 5 个。
- 对于数据库接口的后端文件中，无会导致数据错误或者一致性丧失的代码缺陷。

### 3.3.6 <错误>

- 非致命性错误：包括前端样式错误、因为代码导致的通信滞后（不影响数据一致性）和其他的不影响网站的功能实施和用户主要体验和操作逻辑实现的错误类型；
- 严重错误：用户个人数据完全丢失或完全无法使用应用的某个功能；
- 引入性错误：因为项目开发中需要引入包括前端框架和数据中间件等多种第三方开发的模块，我们对这些模块的代码操控和优化能力有限，这些模块中可能会引入一些错误，我们需要保证这些错误不会影响项目的正常实现。

## 3.4 性能

### 3.4.1 <对事务的响应时间>

对事务的平均响应时间应不超过 0.8s，最长响应时长不应超过 5s，对于超过响应时间限制的事务请求，定位原因（比如是数据库的原因或者网络的原因），并反馈给用户以可读的提示，终止该事务的请求。

### 3.4.2 <吞吐量>

每秒处理的事务数应不少于 100，对于请求资源过多的事务，设置额外的队列进行处理，避免垄断网络的资源通道。

### 3.4.3 <容量>

系统可以容纳的事务数应不少于 1000。

### 3.4.4 <降级模式>

系统在维护期间，用户只能搜索阅读论文，采用模块化编程，方便对子系统功能进行更改

### 3.4.5 <资源利用情况>

资源可利用情况分为两个阶段：

- 本地测试阶段：当工程在本地进行测试的阶段中，使用小组成员的个人设备作为测试平台，可用资源约为 2G 内存和 10G 的 PCIe 接口固态硬盘设备；
- 线上测试阶段：当工程被部署到了在线的服务器上后，可使用的资源视对应的设备而定，原则上不会低于本地测试阶段的可用资源。

## 3.5 可支持性

### 3.5.1 <编码标准>

后端基于 java1.8，前端使用 vue2.0 框架，数据库使用 MySQL5.7

### 3.5.2 <命名约定>

- 对于变量的命名应避免使用抽象或含糊不清的命名方式，如“a”、“something”，而应采用

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

具体并直观的命名

- 名字中每个单词的首字母（比如类型或变量）要大写，单词之间通常不要使用下划线。比如，Food 和 StudentName 是符合规范的，而 studentName 和 student\_name 是不符合规范的
- 类型名称有一个额外的大写字母前缀，以便将类型名称和变量名称区分开来。比如 FSkin 是类型名称，而 Skin 是 FSkin 的一个实例
- 类型和变量名称必须是名词，方法名称是动词，例如 FetchPapers 可以作为一个合适的方法的命名
- 所有返回布尔值的函数都应该询问返回值是真是假这个问题比如 “IsStudent()” 或 “ShouldClearInfo()”。所有布尔变量都必须以 “b” 字母为前缀
- 过程函数命名应该使用一个动词加一个动作对象，减少编码时出现不清楚函数处理对象的情况

### 3.5.3 <访问权限>

对于网站的访问限制，我们设计了多层次的结构，不同层级上的用户可以享用不同级别的访问权限：

- 普通用户具有设置个人信息、查询浏览论文及收藏评论论文等权限
- 管理员除了上述功能外，还可以发布系统公告、更新论文库、修改用户权限等
- 包括网站开发者和授权的技术人员可以直接访问网站的后台代码和数据库内容，并对包括管理员身份内重要的网站数据库内容进行编辑。

## 3.6 设计约束

### 3.6.1 <设计约束一>

网站前端基于 Vue.js 框架完成，后端依赖 java 语言的相关框架（比如 Hibernate、Spring 等），数据库选用标准 MySQL 数据库，后端的数据获取和修改基于 Hibernate 完成。

### 3.6.2 <设计约束二>

项目的开发过程基本遵循以下流程设计：

- a) 网站功能设计，应用场景说明和开发可行性验证；
- b) 开发计划细化、确定开发的技术路径、技术框架和工具、网站接口设计，功能性和非功能性需求的确定
- c) 开发进度表制定以及根据不同成员的意向和技术特点，分配开发任务，在开发过程中，对任务的进度进行每日站立讨论和每周以及阶段性的任务总结和调整
- d) 在两周内得到可以进行主要功能性需求测试的原型程序，进行测试
- e) 继续优化网站设计和功能性功能，对非功能性需求特点进行补充开发，建立测试样例库，编写测试程序，利用自动化部署和测试工具进行集成开发
- f) 第四周得到完整版本，在自己测试的基础上邀请其他用户进行试用，收集测试信息。同时，编写系统的使用说明和用户手册
- g) 进行最后的部署前修改，对之前反馈的不足和错误进行完善
- h) 对系统进行在线部署

## 3.7 接口

### 3.7.1 用户界面

因为网站项目还没有进入实际的开发阶段，暂时不提供界面的预览效果。但简要列出项目中需要实现的界面列表：

<项目名称>	Version: <1.0>
软件需求规约	Date: <dd/mmm/yy>

3.7.1.1 首页

3.7.1.2 搜索页

3.7.1.3 注册登录页

3.7.1.4 个人信息页

3.7.1.5 论文详情页

3.7.2 硬件接口

无

3.7.3 软件接口

Web 后端通过 Hibernate 与 MySql 进行交互

3.7.4 通信接口

用户通过 Http 协议访问 Web 应用。